The BitTorrent Engineering Blog

Notes from the front lines of BitTorrent innovation

DHT Bootstrap Update

arvid — December 19, 2013 — 2 Comments

Arvid Norberg, chief architect for BitTorrent, Inc, introduces a new DHT bootstrap server. This latest version introduces Node ID enforcement as an important step in our development for BitTorrent Chat. It's also now open source so that anyone can run their own bootstrap node.

The BitTorrent Distributed Hash Table (DHT) has a fundamental dependency on being introduced to some nodes that are already in the network. There are many sources of these nodes. For instance, your client is likely to save nodes on disk to retry them when you start back up again. Any BitTorrent peers are likely to be on the DHT as well, so those are also tried. However, if you just installed a BitTorrent client, and you don't have any BitTorrent peers, you must rely on a bootstrap server.

BitTorrent Inc. runs ``router.bittorrent.com`` on port 8991 for this purpose.

We are now providing our DHT bootstrap server open source on github. You can now run your own DHT bootstrap node! Please play with it and contribute fixes, features, and performance improvements.

The DHT bootstrapper has some interesting properties. Up until 5 years ago or so, ``router.bittorrent.com`` was running just another DHT node, just like the one in µTorrent. This had some obvious problems. Since the default routing table size is 8 nodes per bucket, half of all requests to the bootstrap would get the same 8 nodes handed back to it. At several thousand requests per second, this would effectively DDoS any poor node that happened to end up in its routing table.

We rewrote the bootstrap server to have a flat array of nodes instead and to have two cursors, one for reading and one for writing new nodes into it. Every node that pings the bootstrap server is put in a queue and pulled out 15 minutes later to be pinged. If it is still alive, it is added to the node list.

This is still the case with the latest rewrite, with one addition: Node ID enforcement. We have been looking at securing the DHT, making it harder to attack (especially with sybils). One thing we're implementing to support this is requiring DHT nodes to calculate their node ID based on their external IP, with some flexibility to support NATs and such. More info on Node ID enforcement can be found here.

The idea is that with Node ID enforcement sybil attacks, where one machine pretends to be thousands of nodes, will become impossible.

The new bootstrap server will still serve nodes with invalid node IDs (in fact, legitimate nodes just joining are not likely to know their external IP yet). However, it will not ping nor add these nodes to the node list for handing out.

This is one step in the preparations we're making for BitTorrent Chat, which will rely on the DHT and benefits from having a DHT that's harder to eavesdrop and scrape.

Share this: Facebook Twitter Reddit Email

In Announcements

arvid

Posts

Arvid Norberg is the Chief Architect at BitTorrent, Inc., a veteran of the µTorrent team, one of the key members of the BitTorrent Chat development team, and expert in peer-to-peer technology.

Tech Talks: Start Developing... Update on BitTorrent Chat

Ghostery a bloqué les commentaires publiés par Disqus.

Search...

Recent Posts

Tech Talks: C++ in the 21st Century

The BitTorrent Engineering Team: Building Faster Releases

Tech Talks: Writing High-Performance Software

Update on BitTorrent Chat

DHT Bootstrap Update

Recent Comments

Thief on Tech Talks: C++ in the 21st Century

Richard Sandberg on Update on BitTorrent Chat

gaborgabor on Update on BitTorrent Chat

FuturesBrightFuturesCrypto on Update on BitTorrent Chat

FuturesBrightFuturesCrypto on Update on BitTorrent Chat

Archives

March 2014

February 2014

December 2013

November 2013

April 2013

March 2013

February 2013

January 2013

December 2012

Categories

Announcements

BitTorrent Live

Tech Talks

Torque

Uncategorized

µTorrent

Meta

Log in

Entries RSS

Comments RSS

WordPress.org

© 2014 The BitTorrent Engineering Blog